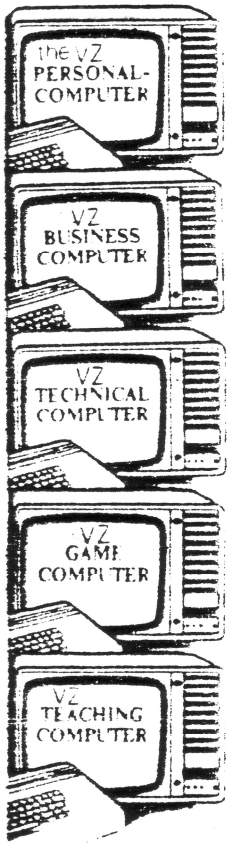
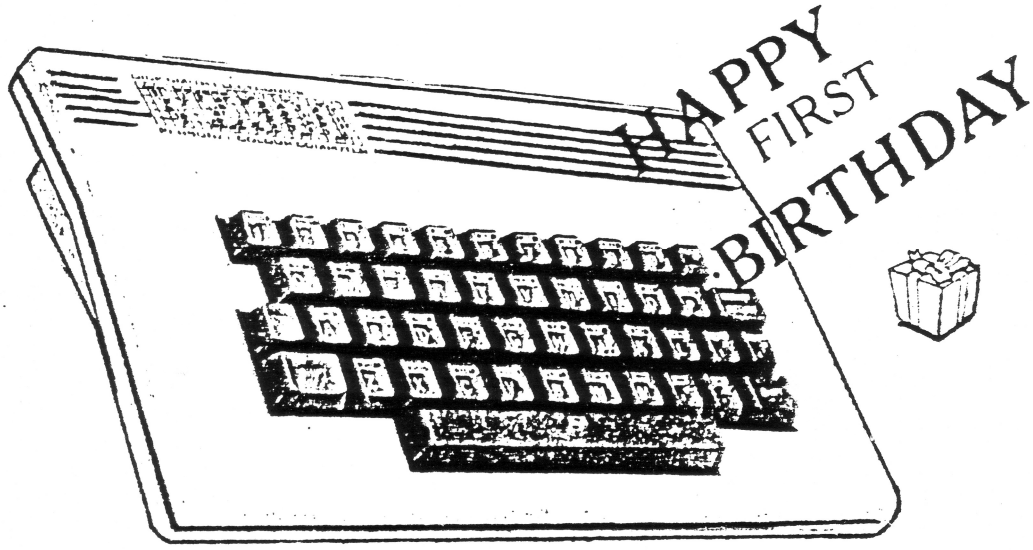
VZ-LINK

NEWSLETTER





AUCKLAND VZ 300 / 200
COMPUTER USERS CLUB
P.O BOX 1972 c.p.o

ISSUE No. 12 - SEPT 1987

Postal address Phone 673-967 before 9 PM please

P.O.BOX 1972 C.P.O AUCKLAND

(c) Auckland VZ 300/200 Computer user's club 1986

All material in this newsletter is subject to copyright Contributed program's& articles can be reproduced on the understanding that they are for the private use of subscriber's only.
Copyright is retained by the author.

CONTRIBUTIONS

The following points will help those thinking of sending programs for publication in this newsletter.

If possible send a listing and a tape of your program. This helps me prepare the newsletter and avoid mistakes. The tape will be returned when i have finished with it.
Include some notes on what the program does and how to use it.
What is painfully obvious to the author is often obscure to we users.

For long or complex program's let me know if i can supply members with a copy of it on tape or if you are prepared to do this yourself.

The auckland users club will not be held responsible for the breeching of copyright caused by the publication of any material sent in under a false name.We like to check that we have permission to reproduce programs and items. and thank those that have given their permission to do so.

I:URGENTLY REQUIRE PROGRAMS AND ITEMS FOR THIS NEWSLETTER.

MEMBERSHIP

Membership $10.00 per year(for 12 issues of newsletters)
This newsletter is available to members only.
New members will receive back issues(if available) free on request

Please make cheques payable to:
VZ 300/200 COMPUTER CLUB

Wow issue number 12 and still going strong ,well i hope you have all had a go at some of the programs from the past issues.
If anyone missed part two of larry Taylors 'toward a vz printer patch' in the aust. electronics monthly june issue please give me a call and i will send you a copy. I now have my disk drive and my VZ now has a new surge of life.I don't know how i lived with out a disk drive.

I was planning the much wanted meeting for october but i'am now looking at a date closer to november .The reason being that my wife will be going into hospital .I am sorry about this and a firm date,venue etc will be in the next issue.I have had a heap of programs etc offered for the newsletter from new members.Please send your items to me and i will use them.In this issue we continue silver mountain(we still have two or more parts to go)after this.If you want a tape or diskcopy of this plus the other public domain software available send a blank disk or tape.We also start a new series written by Robert Quinn called Understanding your VZ,This is going to be a real helpful series for all VZ users.

As you can see i have had a few printing problems this issue ,due to my photocoping contact being of work sick (it is that time of year) If anyone can put me onto someone with a printer for sale please phone. also if someone would like to offer the use of their printer in the mean time it will be a help.I will supply the paper(10")of course. Well on with the programs etc until the next issue best wish's.

Further to our item on the CLEAR command,Peter Patterson has sent us this further explanation of how clear can be used:
CLEAR is also used to set all numeric variables to Zero and erase <u>all</u> data that maybe held by string variables;

Example :

```
20 A=300                'Variable A equals 300
30 A$="test string" 'String Variable denoted with A$
40 Print"before 'clear' A=";A
50 Print "and string variable A$=";A$
60 CLEAR
70 Print "After 'clear' A=";A
80 Print "And string variable A$=";A$
90END
   Type RUN and Press RETURN
```

Before 'CLEAR' command A=300 & string Variable A$="Test string"
After 'CLEAR' command A=Ø and string Variable A$=" (NIL) "

SILVER MOUNTAIN PT2


```
1030 IFR=38ANDF(65)=0,R$="RATS NIBBLE YOUR ANKLES":RETURN
1040 IFR=58AND(D=1ORD=4)ANDF(66)=0,R$="YOU GET CAUGHT IN ":SL=1
1045 IFSL=1,SL=0:R$=R$+" THE WEBS!":RETURN
1050 IFR=48ANDD=4ANDF(70)=0,R$="THE DOOR DOES NOT OPEN":RETURN
1060 IFR=40ANDF(47)=1,F(68)=1
1070 IFR=37ANDD=4ANDE$(37)="EW",R=67:R$="THE PASSAGE WAS ":SL=1
1075 IFSL=1,SL=0:R$=R$+"STEEP!":RETURN
1080 IFR=29ANDD=3,F(48)=1:F(20)=0
1090 IFR=8ANDD=2,F(46)=0
1100 OM=R:FORI=1TOLEN(E$(R))
1110 K$=MID$(E$(OM),I,1)
1120 IF(K$="N"ORK$="U")ANDD=1,R=R-10
1130 IFK$="E"ANDD=2,R=R+1
1140 IF(K$="S"ORK$="D")ANDD=3,R=R+10
1150 IFK$="W"ANDD=4,R=R-1
1160 NEXTI:R$="OK"
1170 IFR=OM,R$="YOU CANNOT GO THAT WAY"
1180 IF((OM=75ANDD=2)OR(OM=76ANDD=4)),R$="OK. YOU CROSSED"
1190 IFF(29)=1,F(39)=F(39)+1
1200 IFF(39)>5ANDF(29)=1,R$="CPPUT IBWF XPSO PVU":GOSUB4260:SL=1
1205 IFSL=1,SL=0:F(29)=0:C(3)=81
1210 RETURN
1220 GOSUB4400:GOSUB3330:R$="OK":F(49)=0
1230 PRINT"YOU HAVE ";
1240 FORI=1TOG:READO$:GOSUB3350:IFI=1ANDC(1)=0ANDF(44)=1,SL=1
1245 IFSL=1,SL=0:O$="COIN"
1250 IFI=GANDC(5)=0,1270
1260 IFC(I)=0,PRINTO$;",";:F(49)=1
1270 NEXTI:IFF(49)=0,PRINT"NOTHING"
1280 PRINT:GOSUB3360:RETURN
1290 IFH=6577,R$="HOW?":RETURN
1300 IFH=4177ORH=5177,B=16:GOSUB2380:RETURN
1310 IFB=38,R$="TOO HEAVY!":RETURN
1320 IFB=4ANDF(43)=0,R$="IT IS FIRMLY NAILED ON!":RETURN
1330 CO=0:FORI=1TOG-1:IFC(I)=0,CO=CO+1
1340 NEXTI:IFCO>13,R$="YOU CANNOT CARRY ANY MORE":RETURN
1350 IFB>G,R$="YOU CANNOT GET THE "+T$:RETURN
1360 IFB=0,RETURN
1370 IFC(B)<>R,R$="IT IS NOT HERE!"
1380 IFF(B)=1,R$="WHAT "+T$+"?"
1390 IFC(B)=0,R$="YOU ALREADY HAVE IT"
1400 IFC(B)=RANDF(B)=0,C(B)=0:R$="YOU HAVE THE "+T$
1410 IFB=28,C(5)=81
1420 IFB=5,C(28)=0
1430 IFC(4)=0ANDC(12)=0ANDC(15)=0,F(54)=1
1440 IFB=8ANDF(30)=1,C(2)=0
1450 IFB=2,F(30)=0
1460 RETURN
1470 R$="YOU SEE WHAT YOU MIGHT EXPECT!"
1480 IFB>0,R$="NOTHING SPECIAL"
1485 IFR=80ANDT$="COTTAGES",R$="YOU CAN SEE A POT":RETURN
1490 IFB=46ORB=88,GOSUB2550
1500 IFH=8076,R$="IT IS EMPTY"
1510 IFH=8080,R$="AHA!":F(1)=0
1520 IFH=7029,R$="OK":F(2)=0
1530 IFB=20,R$="NBUDIFT JO QPDLFU":GOSUB4260:C(26)=0
1540 IFH=1648,R$="THERE ARE SOME LETTERS '"+G$(2)+"'"
1550 IFH=7432,R$="UIFZ BSF BQQMF USFFT":GOSUB4260:F(5)=0
1560 IFH=2134ORH=2187,R$="OK":F(16)=0
1570 IFB=35,R$="IT IS FISHY!":F(17)=0
```

```
1580 IFH=3438,R$="OK":F(22)=0
1590 IFH=242,R$="A FADED INSCRIPTION"
1600 IF(H=14430RH=1485)ANDF(33)=0,R$="B HMJNNFSJOH GSPN U":SL=1
1605 IFSL=1,SL=0:R$=R$+"IF EFQUIT":GOSUB4260
1610 IF(H=14430RH=1485)ANDF(33)=1,R$="SOMETHING HERE...":F(12)=0
1620 IFH=24790RH=2444,R$="THERE IS A HANDLE"
1630 IFB=9,R$="UIF MBCFM SFBET 'QPJTPO'":GOSUB4260
1640 IFH=4055,GOSUB3290
1650 IFH=2969ANDF(48)=1,R$="VERY UGLY!"
1660 IFH=71580RH=7186,R$="THERE ARE LOOSE BRICKS"
1670 IFR=49,R$="VERY INTERESTING!"
1680 IFB=520RB=820RB=81,R$="INTERESTING!"
1690 IFH=6978,R$="THERE IS A WOODEN DOOR"
1700 IFH=6970,R$="YOU FOUND SOMETHING":F(4)=0
1710 IFH=2066,R$="A LARGE CUPBOARD IN THE CORNER"
1720 IFH=68650RH=6853,R$="THERE ARE NINE STONES"
1730 IFH=248,R$="B GBEFE XPSE - 'N S I T'":GOSUB4260
1740 RETURN
1750 IFR=64,R$="HE GIVES IT BACK!"
1760 IFH=6425,GOSUB3210
1770 IFR=750RR=76,R$="HE DOES NOT WANT IT"
1780 IFB=62ANDF(44)=0,R$="YOU HAVE RUN OUT!"
1790 IF(H=75620RH=7662)ANDF(44)>0ANDC(1)=0,R$="HE TAKES IT":SL=1
1795 IFSL=1,SL=0:F(64)=1
1800 IFF(64)=1,F(44)=F(44)-1:IFF(44)=0:C(1)=81
1810 IFB=1,R$="HE TAKES THEM ALL!":C(1)=81:F(64)=1:F(44)=0
1820 IFH=2228ANDC(5)=81,R$=XB$+"NORTH":C(28)=81:R=12
1830 IF(H=2228ANDC(5)=0)ORH=225,R$=XB$+"NORTH":R=12
1840 IF(H=1228ANDC(5)=0)ORH=125,R$=XB$+"SOUTH":R=22
1850 IFR=70RR=33,R$="HE EATS IT!":C(B)=81
1860 IFH=711,F(46)=1:R$="HE IS DISTRACTED"
1870 IFH=3850RH=3824,R$="THEY SCURRY AWAY":C(B)=81:F(65)=1
1880 RETURN
1890 R$="YOU SAID IT"
1900 IFB=84,R$="YOU MUST SAY THEM ONE BY ONE!":RETURN
1910 IFR<>470RB<710RB>750RC(27)<>0,RETURN
1920 IFB=71ANDF(60)=0,R$=X7$:F(60)=1:RETURN
1930 IFB=72ANDF(60)=1ANDF(61)=0,R$=X8$:F(61)=1:RETURN
1940 IFB=(F(52)+73)ANDF(60)=1ANDF(61)=1,F(62)=1:RETURN
1950 R$="THE WRONG SACRED WORD!":F(56)=1:RETURN
1960 IFB=50RB=10,GOSUB1290
1970 RETURN
1980 IFB=3,F(29)=1:R$="ZPV BSF JOWJTJCMF":F(55)=0:GOSUB4260
1990 IFB=20,F(51)=1:R$="ZPV BSF EJTHVJTFE":F(55)=0:GOSUB4260
2000 RETURN
2010 IFB=20RB=14,R$="NOTHING TO TIE IT TO!"
2020 IFH=7214,R$="IT IS TIED":C(14)=72:F(53)=1
2030 IFH=722,R$="OK":F(40)=1:C(2)=72
2040 RETURN
2050 IFH=1547ANDF(38)=1,R$="ALL RIGHT":R=16
2060 IFB=140RB=2,R$="NOT ATTACHED TO ANYTHING!"
2070 IFH=5414ANDC(14)=54,R$="YOU ARE AT THE TOP"
2080 IFH=7214ANDF(53)=1,R$="GOING DOWN":R=71
2090 IFH=722ANDF(40)=1,R=71:R$="IT IS TORN":C(2)=81:F(40)=0
2100 IFH=7114ANDF(53)=1,C(14)=71:F(53)=0:R$="IT FALLS DOWN-BUMP!
2110 RETURN
2120 IFH=522,R$="OK":F(30)=1
2130 IFB=10RB=620RB=50RB=280RB=110RB=24,GOSUB1750
2140 IFH=416,R$="ZPV IBWF LFQU BGMPBU":F(31)=1:GOSUB4260:RETURN
```

UNDERSTANDING YOUR VZ . . . PART ONE.

First things first. Sincere thanks to Mr. Robert Quinn, of Wagga Wagga for providing us with the following article. I'm of understanding that this article was also offered to a leading computer publication. Can't understand why they didn't welcome it with open arms. Oh well! Their loss is certainly our gain. Thanks again Robert, especially for taking time out before your annual pilgrimage (?).

The following is the first of what, at this stage, will be a three part article. Program LLISTings will be publisned as the are relevant to the current section of the article.

VZ MEMORY MAP		TOP OF MEMORY 36863 or 53247 (VZ200) 47103 or 63487 (VZ300)	
30897/8	DISK BUFFER		
30880/1	STRING SPACE		
	BASIC STACK		
30973/4	FREE MEMORY		
30971/2	ARRAY VARIABLES LIST TABLE		
30969/70	SIMPLE VARIABLES LIST TABLE		
30884/5	PROGRAM MEMORY	31465	
	COMMUNICATIONS REGION	30720	745
	VIDEO DISPLAY	28672	2K
	KEYBOARD, CASSETTE SPEAKER, VIDEO CONTROLLER	26624	2K
	ROM CARTRIDGES (DISK CONTROLLER)	16384	10K
	ROM, BASIC INTERPRETER OPERATING SYSTEM	0	16K
ADDRESS POINTERS A/B		START ADDRESS	# of BYTES

ADDRESS = PEEK(A) + PEEK(B) X 256

UNDERSTANDING YOUR VZ . . . PART ONE.

Some addresses in the COMMUNICATION REGION and their use.

30752,30753	pointer to address in video memory of cursor position (range: 28672 to 29183)
30862,30863	pointer to execute address for USR program
30880,30881	pointer to lowest address of string space
30884,30885	pointer to start address of basic program
30887,30888	pointer to start keyboard buffer
30897,30898	pointer to end of user memory
30934,30935	pointer to next available place in string space
30969,30970	pointer to start of variable list table (end of basic program)
30971,30972	pointer to start of array variable table
30973,30974	pointer to end of array variable table
30975,30976	pointer to current address in DATA statement
30882,30883	current line number as basic program runs
30938,30939	line number of last DATA statement READ
30956,30957	line number for last basic line entered/deleted
31009,31010	USR ARGUMENT store
31208	start of INPUT buffer (basic lines and commands)
30776	enable/disable INVERSE printing: POKE 30776,2 enables
30779	copy of output latch/key beep: 0 disables; 1 enables
30876	output device code
31003	TRACE ON/OFF: POKE 31003,1 for on/ 0 for off
30886	cursor (TAB) position
30845,30846,30847	interrupt exit
30977 to 31002	variable declaration table
30720 to 30731	four machine code JUMP instructions to various addresses in ROM

NDERSTANDING YOUR VZ . . . PART ONE.

Here is a collation of understandings to help you exploit the dden facilities, and realise more fully the powers, of your VZ. et's begin with a briefing on some fundamentals as we work our y up the VZ's memory map.

At the bottom of memory is READ ONLY MEMORY -- ROM -- where, rmanently etched byte by byte into their memory cells, reside e intricate sets of machine code instructions and data that ke up the VZ's operating system. Without them, your VZ doesn't ow what to do. with them, it is your obedient servant.

Each memory cell is identified by a unique address, a number u can use to PEEK at the content of that cell. Though, if the dress of a memory cell is greater than 32767, you must use a gative address for PEEKs and POKEs. This negative address is e ordinary address less 65536.

Eg. PRINT PEEK(42326-65536).

A cell is filled with a "number", the BYTE, which can be any nber in the range 0 to 255. You can PEEK (take a look) at ROM ls, but you can't POKE/change their contents. VZ ROM occupies dress from 0 to 16383. The next 10240 (10K:1K = 1024 bytes), nory cells may or may not exist for your VZ; these addresses e reserved for "PLUG IN" memory cartridges. If you have a disk ve, this is where the ROM memory in your disk controller takes residence in your VZ.

Where ROM ends, RAM begins! And it is RAM memory that will our main concern from here on. You can both PEEK at (what's ere?) and POKE to (take that!) RAM memory cells, with the tainty that the POKEs will change the contents of the cells. l memory begins at address 26624 and ends - who knows? It all pends on your system. The built in memory of a VZ 200 (VZ 300) l end at 36863 (47103). A VZ 200 (VZ 300) with 16K memory pansion module installed will end at 53247 (63487). You can d the END OF MEMORY address with the following command :-

PRINT PEEK(30897) + PEEK(30898) * 256

With a disk drive installed, the disk controller reserves 311 ory cells at the top of memory for its own use, and the end of ory command will PRINT an address that is 311 bytes less than actual end of memory address. The disk controller does

HI RES GRAPHICS GEOMETRIC PLOTTING.

(A PLEA FOR MORE READABLE BASIC PROGRAMS).

The following program is a simple line plotting routine using
res graphics screen. It was written to try and demonstrate how
ming skills can be improved by following a few simple guidelines.

Unfortunately published programs in magazines are generally
amples of how to develope good programming style. A number of
have taken the trouble to to enter a listing from a magazine -
n running the program have found that all is not well with the
long, tedious and frustrating session-of understanding the
constructed code, determining all the twists and turns of the
l spaghetti' and debugging-commences. A usual remedy is to
e the program from scratch. Not a very efficient process!

The program below is
learly coded and set out - an enormous help in UNDERSTANDING.
he program is STRUCTURED - a good algorithm is selected and the
program 'flows' through initialization
to input, procedure and output sections.
oops are indented for ease of identification and nesting.
aming of variables is meaningful to assist maintenance and debugging.
nteger storage is used where appropriate.
o abbreviated forms of BASIC statements are used.
emarks are liberally sprinkled throughout to aid clarity.
rror capture and range checking on all input variables prevents
rogram from crashing.

Clear readable code is more important than the execution speed
age requirements of the program - interpreted BASIC runs like a
nail in any case!

These guidelines should lead to code that is easier to read,
and and debug. This leads to easier maintenance, updating or
on of your routines as your programming skills develope.

```
0 REM ******************************
0 REM PLOT A SET OF UP TO 20 LINES     Introduction to program,
0 REM USING THE HI-RES SCREEN.         version and author.
0 REM R.B.KITCH 22/10/85
0 REM ******************************
0 REM DIM STORAGE VECTORS X% & Y%
0 DIM X%(20), Y%(20)                   Vectors to hold end coordinates
0 REM ***ACCEPT INPUT AND CHECK*****   of LN% lines - LN%+1 points.
0 PRINT"HOW MANY LINES - MAX 20":
  INPUT LN%
0 IF LN%<1 OR LN%>20 THEN GO TO 130    Test input is not over-ranged.
0 FOR I% = 0 TO LN%                    Loop for LN%+1 X-Y points.
0     PRINT"ENTER X-VAL  0-127":
      INPUT X%(I%)
0     IF X%(I%)<0 OR X%(I%)>127        Check value not off screen.
      THEN GO TO 160
0     PRINT"ENTER Y-VAL  0-63":
      INPUT Y%(I%)
0     IF Y%(I%)<0 OR Y%(I%)>63         Check value not off screen.
      THEN GO TO 180
0 NEXT I%                              End of input loop.
```

```
300 REM***SET UP SCREEN AND MAIN LOOP*
310 MODE(1)                            Switch screen to hi-res.
320 FOR I% = 0 TO LN%-1                Initialize main loop for lines.
330     X1%=X%(I%):X2%=X%(I%+1)        Assign end points of line to
340     Y1%=Y%(I%):Y2%=Y%(I%+1)        temporary variables.
350 REM ***ARE POINTS THE SAME?*******
360     IF X1%<>X2% OR Y1%<>Y2% THEN   End points the same so PLOT
        GO TO 410                      point.
370     SET(X1%,Y1%):GO TO 710         Pick up another line.
400 REM ***CALC X AND Y DIFFERENCE****
410     DX%=X2%-X1%:DY%=Y2%-Y1%        Change in X and Y directions.
420 REM ***SEE WHICH IS LARGER********
430     IF ABS(DX%)>ABS(DY%)THEN       Branch according to which
        GO TO 610                      difference is larger.
500 REM ***INCREMENT IY***************  Increment along Y-axis.
510     YS%=SGN(DY%):DG=DX%/DY%        Sign of STEP and GRADIENT.
520     XO=X1%+0.5                     X-axis OFFSET.
530     FOR IY% = Y1% TO Y2% STEP YS%  Initialize loop.
540          TP=(IY%-Y1%)*DG+XO        Temporary real X-value.
550          IX%=INT(TP)               Integer X-value.
560          SET(IX%,IY%)              PLOT point.
570     NEXT IY%                       END loop.
580 GO TO 710                          Pick up another line.
600 REM***INCREMENT IX****************  Increment along X-axis.
610     XS%=SGN(DX%):DG=DY%/DX%        Sign of STEP and GRADIENT.
620     YO=Y1%+0.5                     Y-axis OFFSET.
630     FOR IX% = X1% TO X2% STEP XS%  Initialize loop.
640          TP=(IX%-X1%)*DG+YO        Temporary real Y-value.
650          IY%=INT(TP)               Integer Y-value.
660          SET(IX%,IY%)              PLOT point.
670     NEXT IX%                       END loop.
700 REM***END LOOP FOR LINE***********
710 NEXT I%:SOUND 0,9                  END main loop and PAUSE.
800 REM ***GO AGAIN?******************
810 PRINT" (E) TO EXIT"                Screen message or MENU.
820 PRINT" (P) TO PLOT AGAIN"
830 PRINT" (N) FOR NEW POINTS"
840 INPUT AN$                          Accept response.
850 AN$=LEFT$(AN$,1)                   Accept leftmost character.
860 IF AN$="E"THEN STOP                Logical end of program.
870 IF AN$="P"THEN GO TO 310           Go back and PLOT again.
880 IF AN$="N"THEN GO TO 130           Go back for more input.
890 GO TO 810                          Wrong response.
900 END                                Physical end of program.
```

Lines 300-710 are a general purpose line plotting routine similar
to the PLOT command on a MICROBEE.

Bob Kitch, 7 Eurella St., KENMORE, Qld., 4069 (07)378-3745

```
10 REM        "LOTTO PRINTER"
20 '          """"""""""""""
30 '
40 ' BY K. WRIGHT, 26-JUL-87
50 ' VN. 1.1
60 '
70 DIM Ax(6,7),Bx(10,6),Sx(42),C$(10):U$
="####":V$="#####"
80  DATA "FIRST","SECOND","THIRD","FOURT
H","FIFTH"
90 DATA "SIXTH","SEVENTH","EIGHTH","NINT
H","TENTH"
100 CLS:PRINT @ 257,"HOW MANY BOARDS DO
YOU WANT",,:INPUT B
110 CLS:IF B<4 OR B>10,PRINT @ 263,"BETW
EEN 4 AND 10"ELSE 130
120 SOUND 6,3;2,7;0,9:GOTO 100
130 PRINT @ 353,"DO YOU WANT TO USE A PR
INTER ?
140 PRINT TAB(14);"(Y/N)
150 IF INKEY$<>"",150
160 K$=INKEY$:K$=INKEY$:X=RND(5):IF K$="
",160
170 HC=0:IF K$="Y",HC=1
180 CLS:PRINT @ 130,"THINKING, ONE MOMEN
T PLEASE
190 FOR X=1 TO 10:FOR Y=1 TO 6:Bx(X,Y)=0
:NEXT Y:NEXT X
200 RESTORE:FOR L=1 TO B:PRINT@300,L
210 FOR Q=1 TO 40:Sx(Q)=0:NEXT Q
220 Z=0:FOR X=1 TO 6:FOR Y=1 TO 7:Z=Z+1:
Ax(X,Y)=Z
230 NEXT Y:NEXT X
240 Ax(6,6)=-5:Ax(6,7)=-5
250 FOR N=1 TO 6
260 Z=RND(40):IF Sx(Z)=-1,260
270 Sx(Z)=-1:Bx(L,N)=Z
280 PRINT @ 364,Z:NEXT N
290 READ C$(L)
300 REM: NUMBERS HAVE NOW BEEN
310 'CALCULATED,AND ARE STORED
320 'IN ARRAY Sx(?) AND Bx(?,?)
330 'SO SHIFT THEM INTO ARRAY
340 'Ax(?,?) AND PRINT.
350 Z=0:FOR X=1 TO 6:FOR Y=1 TO 7:Z=Z+1:
Ax(X,Y)=Sx(Z)
360 NEXT Y:NEXT X:CLS
370 PRINT C$(L);" CARD":IF HC=1,LPRINT C
HR$(14);C$(L);" CARD"
380 FOR X=1 TO 6:FOR Y=1 TO 7
390 IF Ax(X,Y)>0,PRINT USING U$;Ax(X,Y);
 ELSE 410
400 IF HC=1,LPRINT USING U$;Ax(X,Y);:GOT
O 450 ELSE GOTO 450
410 IF Ax(X,Y)=-1,PRINT"  ";CHR$(224);CH
R$(224);ELSE 430
420 IF HC=1,LPRINT"   ";CHR$(127);CHR$(1
27);:GOTO450ELSE450
430 PRINT "  ";CHR$(L+64);" ";
440 IF HC=1,LPRINT "   ";CHR$(14);CHR$(L
+64);CHR$(15);
450 NEXT Y:PRINT:IF HC=1,LPRINT
460 NEXT X:PRINT:PRINT:IF HC=1,LPRINT:LP
RINT
470 IF HC=0,PRINT @ 482,"<SPACE> FOR MOR
E :-";ELSE 490
480 K$=INKEY$:K$=INKEY$:IF K$<>" ",480
490 CLS:NEXT L
500 PRINT:IFHC=1,LPRINT:LPRINTCHR$(14);"
^^^^^^^^^^^^^^^^^^^^^^"
510 IFHC=1,LPRINT"CAVEAT EMPTOR,NIL DESP
ERANDO CARBORUNDUM"
520 PRINT"CAVEAT EMPTOR,NIL DESPERANDO
  CARBORUNDUM ! !"
530 REM DONE
540 FORZZ=1TO2500:NEXT:RUN
```


HARDWARE FOR SALE:

SPEECH SYNTHESISER:

This unit plugs into your PRINTER INTERFACE SOCKET. You can select external power supply, or with a small modification to your printer interface the power is supplied by the VZ itself. This eliminates the need for another power-pack.
The module comes completely built and tested, housed in a plastic case with the Centronics socket fitted. Also includes internal speaker in the case, switchable to an external speaker or amplifier.
Also comes complete with software on tape consisting of Phase Composer and Dictionary to allow for initial testing. The software output routines enable you to get the module up and yapping at you from the word go. Unit comes with an instruction manual.

PRICE $95.00

Write to:-
VZ DOWN UNDER. Scott Le Brun, 5 Cameron Court,
WANTIRNA. VIC. 3152. AUSTRALIA.

E?F W.P.PATCH

This single Patch will convert your EMF TAPE WORD PROCESSOR for full DISK use while retaining all TAPE functions. It can be used with 1 or 2 DRIVES. Below are the two Menus.

E)DIT TEXT	L)OAD
C)LEAR TEXT	S)AVE
P)RINT TEXT	D)IR
L)OAD FILE	E)RA
S)AVE FILE	R)EN
V)ERIFY FILE	I)NIT
Q)UIT PROGRAM	1-2) DRIVE 1
D)ISK MENU	M)ENU

An option for invisible file names is also provided so your sensitive files are protected. A 16K RAM PACK is needed for VZ200/300. The price is $10.00, while for NZ $12.00 AUS and is available from :-
HUNTER VALLEY VZ USERS' GROUP P.O.BOX 161 JESMOND 2299
N.S.W. AUSTRALIA Phone (049)51 2756

VZ GAMERS TOP SCORE LIST

Any VZ game is eligible to be entered on this list,as is any VZ game player who has a highscore to register.

To register your high score you must verify your score in the manner listed below.

On a sheet of paper send in your highscore along with the following details

: Your name
: Game name
: Frame number and Score
: Date

and your signature

If possible get a witness to sign also.Or send a photo to verify score.
This list rely's on your honesty.

Game Name	Score	Frame #	Initials of scorer
Hamburger sam	100	2	pjh
Ghost Hunter	12470	4	PJH

VZ SOFTWARE HIRE LIBRARY

To hire a program from the Available list all you do is send in a request along with $5-00 .Your tape is sent out as soon as possible.
The cost of $5-00 will cover postage and replacement of blank tape should program get lost or damaged.On return of the program tape in a undamaged condition $3-00 will be credited toward your next tape hire.There after you will only need to send $2-00 to cover postage to hire your next tape.
Again this service will depend on your honesty to make it work.

Available program list

Match box
Dynasty derby
Blackjack
Speed reading
Flashword A and B
Hoppy
Elementry geometry

Tennis and Golf Lesson
Discounted cash flow model
Statistics 2
Biorhythm pairmatching/calender
Electronic slot machine/roulette
Portfolio management
Ghost hunter

More titles to be added